Towards Mathematical Sciences

Vol. 2, No. 1, Spring and Summer 2022, pp. 80–90

DOI.10.22067/tmsj.2022.42621

Expository Article

LINEAR ALGEBRA AND TECHNOLOGY

FARIBA NOOHI[1]

[1]Department of Computer, Alzahra Technical and Vocational Institute,
Technical and Vocational University of Khorasan Razavi, Mashhad, Iran
f.noohi@gmail.com

Abstract. In this article we give a breif history of the applications of linear algebra in thechnology. We investigate the infulence of some introductory theorms on our today's world.

2020 Mathematics Subject Classification. 01A60, 01A61

Keywords. Linear algebra, internet, data science, image processing

Date: Rceived 22-4-2022 Revised 2-6-2022 Accepted 7-6-2022 Available Online 10-7-2022.

دوره ۲، شماره ۱، بهار و تابستان ۱۴۰۱، صص. ۹۰–۸۰
DOI. 10.22067/tmsj.2022.42621
مقاله علمی-ترویجی

https://tmsj.um.ac.ir

جبرخطی و فن‌آوری

فریبا نوحی[۱]

[۱] گروه کامپیوتر، آموزشکده فنی و حرفه‌ای الزهرا، دانشگاه فنی و حرفه‌ای خراسان رضوی، مشهد، ایران
f.noohi@gmail.com

چکیده. در این مقاله با زبان ساده تاریخچه‌ای از کاربردهای جبرخطی را در فن‌آوری روز جهان توضیح می‌دهیم. هدف آن است که با ذکر مثال‌هایی نشان دهیم ریاضیاتی که به نظر مقدماتی می‌رسد تا چه حد بر آنچه در دنیای ما می گذرد تاثیر داشته است.

۱. پیش‌گفتار

ریاضیات به ما کمک کرده است تا جهان را بشناسیم و یک نظم فکری را پایه‌گذاری کنیم. این علم قادر است نحوه استدلال منطقی، تفکر نقاد و خلاقانه، توانایی حل مسأله و حتی مهارت برقراری ارتباط را به ما بیاموزد. ریاضی ابزار قدرتمندی است برای درک و نظم بخشیدن به زندگی. ریاضیات در جای‌جای زندگی ما حضور دارد، در کارهایی مانند اندازه گیری، شمردن، اقتصاد، مهندسی و نرم‌افزارها و همچنین در طبیعت، حتی جایی که خود ما حضور نداریم. ریاضی می‌تواند به توضیح پدیده های طبیعی بپردازد و حتی احساساتی مانند درک ما از زیبایی را توصیف کند. به قول توماس بریتز استاد ریاضی دانشگاه نیو ساوث ولز استرالیا «نه تنها ریاضیات زیباست، بلکه زیبایی نیز نوعی ریاضی است و این دو در هم تنیده هستند.» دکتر بریتز زیبایی را در فرآیند ریاضی یافته است. او می افزاید: «از دیدگاه من، ریاضیات بسیار جالب است. از وقتی کودک بودم آن را دوست داشتم. گاهی، زیبایی و لذت ریاضی در مفاهیم، گاهی در نتایج و گاهی در تفاسیر آن است. سایر

2020 Mathematics Subject Classification. 01A60, 01A61

واژگان کلیدی. جبر خطی، اینترنت، علم داده، پردازش تصویر.
تاریخ: دریافت ۱۴۰۱/۲/۲ بازنگری ۱۴۰۱/۳/۱۲ پذیرش ۱۴۰۱/۳/۱۷ انتشار برخط ۱۴۰۱/۴/۱۹
نحوه ارجاع به این مقاله: ف. نوحی، جبر خطی و فن‌آوری، به سوی علوم ریاضی، ۲ (۱۴۰۱)، شماره ۱، ۹۰–۸۰ .

اوقات، این فرآیندهای فکری است که ذهن شمار را باز می‌کند، احساسی که طی این فرآیند می‌گیرید یا تنها کاری که در جریان است؛ مانند گم شدن در یک کتاب خوب» [۱].

با این مقدمه و ستایش ریاضیات به گوشه‌ای از تاریخچه حضور بی‌بدیل آن در تعدادی از عملکردهای روزانه و بسیار متداولی که امروزه همگان با آن در ارتباط هستیم می‌پردازیم.

امروز زندگی ما بدون اینترنت معنا ندارد. ارتباط ریاضیات و اینترنت مانند ارتباط بین یک نویسنده و زبان است. کار یک نویسنده بدون زبان به سرانجام نمی‌رسد و نوشته‌هایش با تکامل زبان در ارتباط تنگاتنگ است. رایانه‌ها به کمک زبان ریاضی به وجود آمدند و دستگاه اعداد در مبنای دو امکان انجام محاسبات، پخش موسیقی، نمایش تصاویر و هر آنچه که امروز در استفاده از اینترنت مشاهده می‌کنیم را فراهم آورده است. همه این‌ها با دو حرف الفبای ۰ و ۱ انجام می‌شود. قوانین منطق ریاضی عملکرد رایانه‌ها، آدرس‌دهی اینترنتی و موتورهای جستجو را مدیریت می‌کنند. همچنین مدیریت داده‌ها و طراحی روش‌های امنیت شبکه نیز با استفاده از ریاضیات انجام می‌پذیرند. از سوی دیگر اینترنت و شبکه وب نیز زمینه مساعدی برای پیشرفت ریاضیات فراهم کرده‌اند و ریاضیدانان با استفاده از امکانات جستجو به آخرین دستاوردهای همکاران خود دسترسی دارند و از طریق ارتباط اینترنتی می‌توانند به تبادل نظر بپردازند. در واقع این معامله‌ای است که برای هر دو طرف سودآور است.

از سوی دیگر دنیای امروز دنیای داده‌هاست. داده‌هایی که قسمت اعظم آن در اینترنت تولید می‌شود و پردازش و استخراج اطلاعات از آنها جزء جدایی ناپذیری از حوزه هایی مانند اقتصاد، امنیت، مطالعات اجتماعی و مانند آن است. بسیاری علوم دیگر نیز در دنیای امروزی ما و با توجه به نیازها شکل گرفته اند. در سراسر شهرها دوربین‌های مدار بسته برای کاربردهای مختلف جایگذاری شده‌اند و در هر لحظه تصاویری به مراکز مختلف ارسال می‌کنند.

یکی از شاخه‌های جذاب ریاضی جبرخطی است. شاخه‌ای که با فضاهای برداری و نگاشتهای خطی بین آنها سروکار دارد. در واقع در حیطه علم امروز جایی وجود ندارد که ردپایی از جبر خطی در آن پیدا نشده باشد. در این مقاله سعی داریم با زبان ساده برخی کاربردهای جبرخطی را در حوزه فن‌آوری به‌خصوص سیستم وب، پردازش داده و فشرده سازی تصاویر بیان کنیم. در اینجا هدف توضیح ریاضی مسأله نیست و به چند و چون و چرایی رسیدن به الگوریتم‌های مورد بحث نمی‌پردازیم، حتی هدف توضیح آخرین دستاوردهای فن آوری و تاثیر ریاضی بر آنها نیست، بلکه تنها با ذکر مثال‌هایی نشان می‌دهیم که قضایای مقدماتی جبرخطی تا چه حد در پیشرفت فن‌آوری سهم دارند و در نتیجه بر زندگی روزمره ما تاثیر گذاشته‌اند. الگوریتم‌های جستجویی که در زیر به آنها می‌پردازیم از اولین الگوریتم‌های طراحی شده به سبک امروزی بودند که در طول زمان تغییر کرده و به الگوریتم هایی بسیار کارآمدتر تبدیل شده اند، ولی سادگی ایده آنها شروع خوبی برای درک کارکرد موتورهای جستجو است.

۲. استفاده از جبر خطی در الگوریتم‌های جستجو

در حوزه فن‌آوری معمولا فضاهایی که متغیرها و داده‌ها در آن قرار می‌گیرند، فضاهای برداری هستند. در نتیجه پردازش مسائل با استفاده از ابزارهای جبر خطی میسر می‌گردد. تقریباً تمام موتورهای جستجوی اصلی وب امروزی از تحلیل پیوند[۱] برای بهبود نتایج جستجوی خود استفاده می‌کنند. تحلیل پیوند نوعی تحلیل داده است که چگونگی ارتباط صفحات وب را تحلیل می‌کند. بنیان این روش و استفاده از ساختار ابرپیوند[۲] وب، برمبنای نظریه ماتریس ساخته شده است. روش تحلیل‌پیوند و پشتوانه‌ی آن یعنی جبرخطی به انقلابی در جستجوی وب منجر شد، به طوری که روش جستجوی پیش از روش تحلیل پیوند (قبل از ۱۹۹۸) در مقابل جستجوی با دقت قابل توجه پس از آن کاملا رنگ باخت.

وقتی کاربری در جستجوی عبارت یا مطلبی است، از موتور جستجو انتظار دارد که صفحات مربوط با آن مطلب را به ترتیب اهمیت در دسترس وی قرار دهد. در حال حاضر وقتی در گوگل چیزی را جستجو می‌کنید این انتظار تا حد بسیار زیادی برآورده می شود. این قابلیت ناشی از وجود الگوریتم‌های جستجوی پیشرفته است که بر روش تحلیل پیوند استوارند. برای این که به اهمیت روش تحلیل پیوند پی‌ببریم برای یک دقیقه وضعیت جستجوی قبل از سال ۱۹۹۸را در نظر بیاورید. به دلیل تعداد بی‌شمار صفحات وب یک پرس و جو از یک موتور اغلب فهرست بسیار طولانی از صفحات، حتی گاهی تا هزاران صفحه مرتبط را نشان می‌داد. کاربر می‌بایست این لیست بلند بالا را مرتب کرده و صفحاتی با بیشترین ارتباط را می‌یافت. واضح است که به دلیل وجود هرزنامه‌ها[۳] ترتیب ارائه شده در نتیجه‌ی جستجو کمک کمی به کاربر می‌کرد، زیرا برای فریب یک موتور جستجو جهت تولید رتبه‌ی بالاتر از حد معمول برای یک صفحه، هرزنگارها[۴] به طور آزادانه از متاتگ‌ها استفاده می‌کردند و ادعا می‌کردند که صفحه آنها از عبارات جستجوی محبوبی برخوردار است، که البته این عبارات هرگز در صفحات ظاهر نمی‌شدند. در اینجا شاید اشاره به این نکته مفید باشد که متاتگ‌ها عبارات کوتاهی در توصیف یک صفحه هستند که در خود صفحه ظاهر نمی‌شوند بلکه در کد آن صفحه وجود دارند. هرزنگارها همچنین عبارات جستجوی محبوب را در متون نامرئی (متن سفید روی زمینه‌ی سفید) بارها تکرار می‌کردند تا موتور جستجو را بفریبند. به این ترتیب کاربر با ترتیبی ساختگی از صفحات ظاهراً مرتبط با موضوع روبرو می‌شد که در واقع نشان از ارتباط واقعی و بیشتر با موضوع مورد نظر نبود. با ورود الگوریتم‌های جدید، متاتگ‌ها به تدریج در موتورهای جستجو بی‌اثر شدند. الگوریتم‌های هیتس[۵] و رتبه‌صفحه[۶] اولین الگوریتم‌های تحلیل پیوند بودند که در حدود سال ۱۹۹۸ توسعه یافتند و هر دو به‌طور قابل ملاحظه‌ای فرآیند جستجو را

---

[۱] link analysis

[۲] hyperlink

[۳] spams

[۴] spammers

[۵] HITS

[۶] PageRank

بهبود بخشیدند. البته هر دوی این الگوریتم ها در طول زمان جای خود را به الگوریتم های بهتری دادند ولی از آنجا که آشنایی با آنها به درک چگونگی کار الگوریتم‌های جستجو و ریاضیات دخیل در انها کمک می‌کند، این الگوریتم ها را به طور مختصر بررسی می‌کنیم. ابتدا به توصیف ساده‌ای از الگوریتم هیتس می‌پردازیم.

۱.۲. **الگوریتم هیتس.** الگوریتم هیتس یک روش تحلیل پیوند است که جان کلاینبرگ[۷] آن را به منظور تمرکز بر لیست‌های طویل و بدون قاعده جستجوها درصفحات وب ابداع کرد [۵]. او متوجه الگویی در بین صفحات وب شد. برخی صفحات مانند یک کانون [۸] یا پرتال عمل می‌کردند زیرا پیوندهای خروجی زیادی داشتند و برخی دیگر مانند قلمرو[۹] یا منبع یک موضوع عمل می‌کردند، زیرا به عنوان صفحاتی که حاوی اطلاعات مهمی درباره موضوع بودند، ارجاعات یا پیوندهای ورودی بسیاری داشتند. کلاینبرگ دریافت که به‌نظر می‌رسد کانون‌های خوب به سوی قلمروهای خوب اشاره می‌کنند و قلمروهای خوب به وسیله کانون‌های خوب مورد اشاره قرار می‌گیرند. بنابراین او به هر صفحه i یک امتیاز کانونی h_i و یک امتیاز قلمروی a_i اختصاص داد. در واقع به‌ازای هر صفحه i، در تکرار، k امتیاز کانونی $h_i^{(k)}$ و امتیاز قلمروی $a_i^{(k)}$ به شکل زیر تعریف شدند:

$$a_i{}^{(k)} = \sum_{j:e_{ij}\in E} h_i{}^{(k-1)}, \quad h_i{}^{(k)} = \sum_{j:e_{ij}\in E} a_i{}^{(k)}.$$

که در اینجا e_{ij} نمایش یک ابر پیوند از صفحه i به صفحه j است و E مجموعه همه ابرپیوندها است. او برای محاسبه امتیاز یک صفحه با امتیاز یکنواخت شروع کرد، یعنی اگر n تعداد صفحات در یک همسایگی خاص از لیست مورد نظر باشد، قرار داد

$$h_i{}^{(1)} = \frac{1}{n}, \quad a_i{}^{(1)} = \frac{1}{n}.$$

در اینجا منظور از همسایگی تمام صفحات لیست مورد نظر و تمام صفحاتی که از این لیست به آنها اشاره شده یا از آنها به صفحات لیست اشاره شده است، می‌باشد و بسته به نیاز لیست مورد نظر می‌تواند شامل صد یا صدها هزار صفحه باشد. این امتیازها با قاعده تکراری فوق تا زمانی که به یک مقدار ایستا همگرا شوند، محاسبه می‌شوند . با کمک جبر خطی می‌توانیم معادلات فوق را به شکل ماتریسی بنویسیم. فرض کنید h و a بردارهای ستونی به ترتیب حاوی امتیازات کانونی و قلمروی باشند و فرض کنید L ماتریس مجاورت در مجموعه همسایگی باشد، یعنی $L_{ij} = 1$ اگر صفحه i به صفحه j پیوند داشته باشد و در غیر این صورت $L_{ij} = 0$. در این صورت داریم

$$a^{(k)} = L^T h^{(k-1)} \text{ and } \quad h^{(k)} = La^{(k)},$$

---

[۷] Jon Kleinberg

[۸] hub

[۹] authority

و با انجام محاسباتی، به نتایج زیر می‌رسیم

$$a^{(k)} = L^T L a^{(k-1)}$$
$$h^{(k)} = L L^T h^{(k-1)},$$

که در اینجا L^TL ماتریس کانونی[۱۰] و LL^T ماتریس قلمروی[۱۱] نام دارند. وظیفه الگوریتم هیتس حل مسأله بردار ویژه

$$L^T L a = \lambda_1 a, \quad L L^T h = \lambda_1 h$$

خواهد بود، که در آن λ_1 بزرگترین مقدار ویژه L^TL (و LL^T) و a و h بردارهای ویژه نظیر خواهند بود. علاوه بر جبرخطی مقدماتی که در الگوریتم هیتس به کار می رود، مسائل مهمی مانند همگرایی، وجود، یکتایی و محاسبه عددی این امتیازات هم وجود دارند که با استفاده از دانش ریاضی حل می‌شوند.

۲.۲. **الگوریتم رتبه صفحه.** هر دوی ایده گوگل و رتبه صفحه توسط سرگئی برین[۱۲] و لاری پیج[۱۳] زمانی که دانشجوی علوم کامپیوتر دانشگاه استنفورد بودند، شکل گرفت. هرچند امروزه الگوریتم‌های بسیار پیشرفته‌تری ابداع شده اند ولی توصیف الگوریتم رتبه‌صفحه می تواند درک موضوع را ساده‌تر کند. نام این الگوریتم یعنی PageRank هم به کلمه page در web page اشاره دارد و هم نام Lary Page. با یک پرسش شروع می کنیم. گوگل چگونه صفحات وب را جستجو می کند؟ پاسخ این است که گوگل صفحات وب را با کمک رتبه‌بندی صفحات برای یافتن کلمات کلیدی مورد نظر جستجو می‌کند. رتبه‌صفحه چیست؟ رتبه صفحه الگوریتمی بود که توسط جستجوگر گوگل برای رتبه‌بندی صفحات وب در نتایج موتورجستجوی خود استفاده می‌کرد و روشی برای اندازه گیری اهمیت صفحات وب بود [۲،۳]. رتبه صفحه با شمارش تعداد و کیفیت پیوندهای ارجاع به یک صفحه کار می‌کرد تا تخمین تقریبی از اهمیت وب سایت را تعیین کند. فرض اساسی این بود که احتمالاً وب‌سایت‌های مهم‌ترپیوندهای بیشتری از وب سایت‌های دیگردریافت خواهند کرد. برین و پیج روش بازگشتی شبیه به روش کلاینبرگ را استفاده کردند. از نظر آنها یک صفحه مهم بود اگر از سوی صفحات مهم دیگر به آن اشاره شود. یعنی از نظر آنها اهمیت صفحه شما (امتیاز رتبه‌صفحه) مجموع امتیازات رتبه صفحه‌هایی است که به صفحه شما اشاره می‌کنند. برای تعریف ریاضی رتبه یک صفحه، برین و پیج اینگونه استدلال کردند که وقتی یک صفحه مهم به چندین مکان اشاره می‌کند، وزن (رتبه‌ی) آن باید به طور متناسب بین صفحات توزیع شود. به عبارت دیگر، اگریاهو به صفحه وب شما اشاره می کند، این خوب است، اما شما نمی بایست وزن (امتیاز) کامل آن را دریافت کنید زیرا آنها به بسیاری از مکان‌های دیگر هم اشاره می‌کنند. اگر یاهو علاوه بر

---

[۱۰] hub matrix

[۱۱] authority matrix

[۱۲] Sergey Brin

[۱۳] Lary Page

صفحه شما به 999 صفحه‌ی دیگر هم اشاره می‌کند آن گاه شما تنها اعتباری معادل 0.001 رتبه صفحه‌ی یاهو را به دست می‌آورید. آنها با استفاده از این ایده و استدلال‌های ریاضی فرمول بازگشتی زیر را پیشنهاد کردند:

$$r_i^{(k+1)} = \sum_{j \in I_i} \frac{r_j^{(k)}}{|O_j|},$$

که در آن $r_i^{(k)}$ رتبه صفحه i–ام در تکرار k، I_i مجموعه صفحاتی که به صفحه i اشاره می کنند و $|O_j|$ تعداد پیوندهای خروجی صفحه j است. این الگوریتم نیز مانند هیتس از یک رتبه یکنواخت برای همه صفحات شروع می‌کند. یعنی اگر تعداد صفحات n باشد با $r_i^{(0)} = \frac{1}{n}$ شروع کرده و با تکرار آن را اصلاح می‌کند. این فرآیند را می توان به کمک ماتریس‌ها نمایش داد. فرض کنید V_k^T بردار ستونی رتبه‌صفحه در k–امین تکرار باشد. در این صورت می‌توان معادلات فوق برای رتبه‌صفحه را به شکل فشرده زیر نوشت:

$$V_{k+1}^T = HV_k^T,$$

که در آن H ماتریس ابرپیوندی با ستون‌های نرمال شده است (مجموع درایه های هر ستون برابر ۱ است)، یعنی

$$h_{ij} = \begin{cases} \dfrac{1}{|O_j|}, & \text{اگر پیوندی از صفحه } j \text{ به صفحه } i \text{ باشد} \\ 0, & \text{.در غیر این صورت} \end{cases}$$

در اینجا سعی می کنیم با یک مثال ساده مسأله را توضیح دهیم [۶].

پنج نفربا نام‌های A، B، C، D و E را فرض کنید که برخی از آنها همدیگر را می‌شناسند و دوست هستند. ماتریس دوستی زیر را در نظر بگیرید.

$$\text{H=} \begin{pmatrix} & A & B & C & D & E \\ A & 0 & 1 & 1/3 & 0 & 1/2 \\ B & 1/4 & 0 & 1/3 & 0 & 1/2 \\ C & 1/4 & 0 & 0 & 0 & 0 \\ D & 1/4 & 0 & 0 & 0 & 0 \\ E & 1/4 & 0 & 1/3 & 1 & 0 \end{pmatrix}$$

با توجه به ستون‌های هر حرف درواقع این ماتریس نشان می دهد که A با هرچهار نفردیگر دوست است و به دوستی همه‌ی آنها وزن مساوی می دهد، B و D هر کدام فقط یک دوست دارند، C سه دوست دارد و E دو دوست دارد. دقت کنید که وزن دوستی‌ها برابر است. بنابراین امتیاز دوستی برای A از این دیدگاه که چه

کسانی و با چه وزنی دوست او هستند، عبارت است از مجموع سطر A. با این توضیح داریم:

$$F(A)=\frac{11}{6}\quad F(B)=\frac{13}{12}\quad F(C)=\frac{1}{4}$$
$$F(D)=\frac{1}{4}\qquad\qquad F(E)=\frac{19}{12}.$$

بیایید ببینیم در نهایت چه کسی بیشترین امتیاز دوستی را دارد؟ در ابتدا شما به همه امتیاز برابر می‌دهید یعنی $[\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5}]$. حال روش تکراری زیر را به کار می‌بریم

$$V_{n+1}=HV_n,$$

که در آن $V_0=[\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5}]^T$, یعنی ترانهاده ماتریس سطری اولیه. این همان روش توانی است[۱۴] که مقدار ویژه غالب و بردار ویژه خاصی متناظر با آن را به ما خواهد داد. در اینجا منظور از مقدار ویژه غالب، مقدار ویژه‌ای مانند λ است که به ازای هر مقدار ویژه دیگر مانند μ، $|\mu|\leq\lambda$. ثابت می‌شود که بردار ویژه‌ای با درآیه‌های نامنفی متناظر با λ وجود دارد (قضیه ۱.۲ زیر را ببینید).

این بردار ویژه را V می‌نامیم، داریم

$$HV=\lambda V.$$

اگر $V=[v_1,\dots,v_5]$ آنگاه امتیاز نفر i-ام برابر است با λv_i.

اکنون مفهوم دوستی را با پیوند در یک وبگاه جایگزین می‌کنیم.

- وبگاه A دوست وبگاه B است، اگر یک ابرپیوند به وبگاه B داشته باشد.
- گوگل با گشتن در وب متوجه می‌شود که چه کسی دوست چه کسی است.
- تعداد ''دوستان'' یک وبگاه برابر است با تعداد پیوندهایی که در آن وبگاه مشاهده می‌شود.
- ماتریس H در حال گشتن در وب ساخته می‌شود، و پس از آن که ساخته شد مقدار ویژه مورد نظر را محاسبه می‌کنیم تا تقریبی برای رتبه‌صفحه به دست آید.

قضیه زیر که فرم اولیه آن برای ماتریس‌های با درایه‌های اکیدا مثبت توسط پرون[۱۵] و بعداً تعمیمی از آن توسط فروبینیوس[۱۶] ثابت شده، وجود بردار ویژه‌ای با درایه‌های نامنفی را برای مقدار ویژه غالب، تضمین می‌کند.

قضیه ۱.۲. *فرض کنید $H=(h_{ij})$ یک ماتریس مربعی حقیقی با درایه‌های نامنفی باشد. در این صورت مقدار ویژه نامنفی λ برای H وجود دارد که به ازای هر مقدار ویژه دیگر مانند μ، $|\mu|\leq\lambda$ و همچنین بردار ویژه‌ای متناظر با آن با درایه‌های نامنفی وجود دارد. λ یک ریشه ساده چند جمله‌ای مشخصه H است.*

---

[۱۴] power method

[۱۵] Perron

[۱۶] Ferobenius

البته اثبات وجود یک جواب و پیدا کردن جواب در عمل، معمولاً دو مقوله جدا هستند و پیدا کردن روشی که منجر به یافتن جواب شود بسیار مهم است. در واقع روش تکراری فوق گاهی مشکل همگرایی و یا افتادن در دور دارد. برای حل این مشکل برین و پیج به اصلاح این روش پرداخته و با استفاده از ماتریس‌های تصادفی الگوریتم‌های دیگری برای یافتن پاسخ های مناسب ساختند. در اینجا به جزئیات کار نمی‌پردازیم بلکه به طور مختصر متذکر می‌شویم که در این روش‌ها درایه‌های ماتریس H، احتمال وجود پیوند بین دو صفحه خواهند بود که پیدا کردن توزیع این احتمال هم خود از اهمیت برخوردار است و پس از آن بایستی تغییرات مناسبی روی این ماتریس اعمال کرد تا همگرایی جواب را تضمین کند. این نشان می‌دهد که برای حل مسائلی از این نوع دانش ریاضی اساس کار است و با طرح هر مسأله جدید سطح و پیچیدگی دانش مورد نیاز نیز بالاتر خواهد رفت. در واقع امروزه الگوریتم های بسیار پیشرفته‌تر به وجود آمده‌اند که کار جستجو را بسیار موثرتر و سریع‌تر انجام می‌دهند ولی بنیاد همه آنها برقضایای ریاضی استوار است.

۳. جبرخطی در داده‌پردازی

روش تحلیل مولفه‌های اصلی[۱۷]، PCA

در جهان امروز، روزانه داده‌های زیادی در سایت های اجتماعی مختلف تولید می‌شود. در واقع ۹۰ درصد از داده‌های امروزی در سه یا چهار سال گذشته تولید شده است. این داده‌ها برای جمع‌آوری اطلاعات مهم و مرتبط باید به درستی تجزیه و تحلیل شوند. PCA تکنیک کاهش ویژگی برای استخراج ویژگی‌های غالب از داده‌ها است، به این معنی که یک مجموعه بزرگ از متغیرها را طوری به مجموعه کوچکتری کاهش می‌دهد که هنوز هم بیشتر ویژگی‌های مهم حفظ شده باشند. این ویژگی‌های استخراج‌شده به عنوان «مولفه‌های اصلی» شناخته می‌شوند. کاهش تعداد متغیرها می‌تواند به بهای کاهش دقت تمام شود، ولی مزیت این روش این است که کمی از دقت را به عنوان بهایی در ازای ساده‌تر کردن مسأله می پردازد، زیرا پردازش داده‌های کمتر در الگوریتم‌های یادگیری ماشین ساده‌تر و سریع‌تر است و از وارد کردن داده‌های اضافی در محاسبات اجتناب می‌شود. این روش را می‌توان به شرح زیر توصیف کرد [۴]:

مجموعه متغیرهای اولیه یک فضای برداری را تشکیل می‌دهند.

- یک مولفه اصلی ترکیب خطی از متغیرهای اولیه است.
- مولفه‌های اصلی به گونه‌ای استخراج می‌شوند که اولین مولفه اصلی حداکثر واریانس در مجموعه داده را توصیف می‌کند.
- مولفه اصلی دوم سعی می‌کند واریانس باقی مانده‌ی مجموعه داده را توصیف کند و با مؤلفه اصلی اول همبستگی ندارد.

---

[۱۷] Principle Component Analysis

- مولفه اصلی سوم سعی می کند واریانسی را شرح دهد که با دو مولفه اصلی اول توصیف نمی‌شود، وبه همین ترتیب.

اولین مولفه مهمترین است و بعد از آن دومی و سومی و الی آخر. حال چگونه مولفه اصلی را پیدا کنیم؟ از نظر هندسی مولفه‌های اصلی جهت‌هایی در فضای داده هستند که متناظر با مقادیر بیشین واریانس می‌باشند، به بیان دیگر خطوطی که بیشترین اطلاعات داده را دربر دارند. ارتباط بین واریانس و اطلاعات در اینجا اینگونه است که هر چقدر واریانس در طول یک خط بیشتر باشد، پراکندگی داده در طول آن بیشتر است و هرچقدر این پراکندگی بیشتر باشد به این معنی است که اطلاعات بیشتری روی آن قرار دارد. ثابت می‌شود که مولفه‌های اصلی، بردارهای ویژه ماتریس کوواریانس مجموعه‌ی داده است. یعنی ماتریسی که درایه ij آن کوواریانس داده i و داده j باشد. از آنجا که $cov(i,i) = var(i)$ و $cov(i,j) = cov(j,i)$ ماتریس حاصل یک ماتریس متقارن خواهد بود که روی قطر، واریانس هر متغیر را دارد. هر بردار ویژه این ماتریس جهتی را نشان می‌دهد که بیشترین واریانس (اطلاعات) را در بر دارد و مقدار ویژه نظیر نیز مقدار این واریانس را به دست می‌دهد. با توجه به این که اغلب با داده‌های بزرگ سروکار داریم، محاسبه بردارها و مقادیر ویژه این ماتریس‌های بزرگ خود نیاز به الگوریتم‌های مفصلی دارد که با استفاده از قضایا و روش‌های ریاضی این کمیت‌ها را محاسبه می‌کنند.

۴. فشرده سازی تصویر

یک تصویر را می‌توان در یک ماتریس ذخیره کرد به‌طوری که هر درایه آن مقدار پیکسل نظیر آن باشد. به عنوان مثال یک عکس سیاه و سفید می تواند در فرمت PNG ذخیره شود که در آن هر پیکسل مقدار ۰ یا ۱ را دارد. فرض کنید تصویری با اندازه 423×480 پیکسل داشته باشیم. یعنی ماتریسی مانند M با اندازه 423×480 داریم که درایه‌های آن ۰ یا ۱ هستند. این ماتریس را با کمک روش تجزیه مقدار منفرد[۱۸]به صورت $M = UDV^T$ تجزیه می‌کنیم، که در آن U یک ماتریس یکانی 480×480، V یک ماتریس یکانی 423×423 و D یک ماتریس 423×480 است که دارای یک زیر ماتریس قطری $r \times r$ می‌باشد، که به تعداد r از مقادیر منفرد M را روی قطر خود دارد دارد و سایر درایه‌های D صفرند. در اینجا منظور از مقادیر منفرد، ریشه دوم مقادیر ویژه (الزاما نامنفی) ماتریس متقارن M^TM است.

حال فقط با کمک ۳۰ مقدار منفرد اولیه تصویر را بازسازی می‌کنیم. تصویر دوم از نظر کیفی تفاوت زیادی با تصویر اول ندارد ولی با کاهش عناصر روی قطر به ۳۰ تا، به جای $480 \times 423 = 203040$ پیکسل، تعداد 30×480 پیکسل در U و 423×30 تا در V^T داریم. به این ترتیب

$$m = 30(1 + 480 + 423)$$

تعداد کل پیکسل‌ها خواهد بود که حدود 13 درصد مقدار اولیه است. در شکل ۱ دو تصویر (آ) و (ب) را مقایسه کنید. تصور کنید که موسسه‌ای با هزار کارمند هر روز بخواهد با استفاده از دوربین چهره افراد وارد شده

[۱۸] singular value decomposition (SVD)

به محل کار را شناسایی کند. این موسسه باید پایگاه داده‌ای حاوی حداقل چهار عکس از هر نفر داشته باشد، یعنی ۴۰۰۰ عکس. در زمان ورود هر کارمند باید عکسی از او گرفته شود و با آنچه در پایگاه داده ذخیره شده مقایسه گردد. مسلما وقتی از روش فشرده سازی استفاده شده باشد، تعداد پیکسل‌های مقایسه شونده به طور معنی داری کمتر و زمان مقایسه نیز کوتاهتر خواهد بود.



(آ) قبل از فشرده‌سازی



(ب) بعد از فشرده‌سازی

شکل ۱. مقایسه دو تصویر

۵. سخن پایانی

مثال های فوق نمونه کوچکی از کاربردهای جبرخطی در فرآیند توسعه سریع علمی در روزگار ما هستند و این مثال‌ها خود به اشکال گوناگون در جاهای مختلفی نظیر مهندسی، پزشکی، اقتصاد و ... کاربرد دارند. این نشان می‌دهد که اگر پیشرفت‌های تئوری ریاضی نبود، جهان ما شکل دیگری داشت.

مراجع

[۱] م. میرزاخانی، *پارادوکسی نزدیک به جادو*، https://dmath-860.niloblog.com/p/44.

[2] S. Brin and L. Page, *The anatomy of a large-scale hypertextual web search engin,* Computer Networks and ISDN Systems, **33** (1998) 107–117.

[3] S. Brin, L. Page, R. Motwami and Terry Winograd, *The PageRank citation ranking: bringing order to the web,* Technical report, Computer Science Department, Stanford University, 1998.

[4] S.M. Holland, *Principal Components Analysis (PCA),* in: Encyclopedia of Environmental Change, edited by: J.A. Matthews, pp. 1–12, SAGE Publications, 2019. https://doi.org/10.4135/9781446247501.n3114.

[5] J. Kleinberg, *Authoritative sources in a hyperlinked environment,* J. ACM. **46** (1999) 604-632.

[6] R.C. Mittal, *Some applications of linear algebra in computer science*, Lectures in Jaypee Inst. Inform. Tech. 2021.